A New Multistage Approach to Motion and Structure Estimation: From Essential Parameters to Euclidean Motion Via Fundamental Matrix

Zhengyou Zhang

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

A New Multistage Approach to Motion and Structure Estimation: From Essential Parameters to Euclidean Motion Via Fundamental Matrix

Zhengyou Zhang

N° 2910

June 1996

THÈME 3

ISSN 0249-6399

INRIA
SOPHIA ANTIPOLIS

A New Multistage Approach to Motion and Structure Estimation: From Essential Parameters to Euclidean Motion Via Fundamental Matrix

Zhengyou Zhang

Thème 3 — Interaction homme-machine,
images, données, connaissances
Projet Robotvis

Rapport de recherche n°2910 — June 1996 — 36 pages

Abstract: The classical approach to motion and structure estimation problem from two perspective projections consists of two stages: (i) using the 8-point algorithm to estimate the 9 essential parameters defined up to a scale factor, which is a linear estimation problem; (ii) refining the motion estimation based on some statistically optimal criteria, which is a nonlinear estimation problem on a five-dimensional space. Unfortunately, the results obtained using this approach are often not satisfactory, especially when the motion is small or when the observed points are close to a degenerate surface (e.g. plane). The problem is that the second stage is very sensitive to the initial guess, and that it is very difficult to obtain a precise initial estimate from the first stage. This is because we perform a projection of a set of quantities which are estimated in a space of 8 dimensions, much higher than that of the real space which is five-dimensional. We propose in this paper a novel approach by introducing an intermediate stage which consists in estimating a 3×3 matrix defined up to a scale factor by imposing the *zero-determinant constraint* (the matrix has seven independent parameters, and is known as the fundamental matrix). The idea is to *gradually* project parameters estimated in a high dimensional space onto a *slightly lower* space, namely from 8 dimensions to 7 and finally to 5. The proposed approach has been tested with synthetic and real data, and a considerable improvement has been observed for the delicate situations mentioned above. Our conjecture from this work is that the imposition of the constraints arising from projective geometry should be used as an intermediate step in order to obtain reliable 3D Euclidean motion and structure estimation from multiple calibrated images.

Key-words: Motion Analysis, Structure from Motion, Gradual Constraint Enforcing, Multistage Algorithm

(Résumé : tsvp)

Unité de recherche INRIA Sophia-Antipolis
2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex (France)
Téléphone : (33) 93 65 77 77 – Télécopie : (33) 93 65 77 65

Un nouvel algorithme pour l'estimation du mouvement et de la structure: Des paramètres essentiels au mouvement euclidien à travers la matrice fondamentale

Résumé : La technique classique pour estimer le mouvement et la structure à partir de deux projections perspectives se compose de deux étapes : (i) utiliser l'algorithme des 8-points pour estimer de manière linéaire les 9 paramètres essentiels définis à un facteur d'échelle près ; (ii) raffiner l'estimation du mouvement à base d'un critère statistiquement optimal. Le problème (ii) est un problème d'estimation non linéaire sur un espace de 5 dimensions. Malheureusement, les résultats obtenus avec cette technique ne sont souvent pas satisfaisants, surtout quand le mouvement est petit ou quand les points observés sont proches d'une configuration dégénérée (par exemple une surface plane). Un problème important est que la deuxième étape est très sensible à l'estimée initiale et qu'il est très difficile d'obtenir une estimée initiale précise à partir de la première étape par projection. Ceci est principalement dû au fait que nous avons estimé des quantités dans un espace de 8 dimensions, qui est beaucoup plus grand que le vrai qui est de 5 dimensions. Nous proposons dans cet article une nouvelle technique en introduisant une étape intermédiaire qui consiste en une estimation d'une matrice 3×3 définie à un facteur d'échelle près *avec la contrainte de déterminant nul* (donc cette matrice a 7 paramètres indépendants, et est connue sous le nom de la matrice fondamentale). L'idée est de projeter *de manière progressive* les paramètres estimés dans un espace de dimensions élevé sur un espace un peu plus faible. Pour notre application, nous passons progressivement de la dimension 8 à 5, en passant par la dimension 7. La technique proposée a été testée avec des données synthétiques et réelles, et une amélioration considérable a été observée dans les situations délicates mentionnées avant. À partir de ce travail, nous formons une conjecture sur la nécessité de l'estimation de la géométrie épipolaire non calibrée comme étape intermédiaire pour obtenir une estimation fiable du mouvement et de la structure euclidiens à partir d'images multiples calibrées.

Mots-clé : Analyse du mouvement, Structure à partir du mouvement, Renforcement progressif de contraintes, Algorithme à étapes multiples

Contents

1 Introduction

Motion and structure from motion has been of the central interest in Computer Vision since its infancy, and is still an active domain of research. There are a large number of pieces of work reported in the literature in this domain. The reader is referred to [29, 1, 14] for a review. The problem is usually divided into two steps: (i) extract features (usually points or line) and match them between images; (ii) determine motion and structure from corresponding features. The earlier work was mainly on the development of linear algorithms and the existence and uniqueness of solutions [21, 40, 9, 27]. More recently, a number of researchers developed algorithms which are noise-resistant by using a sufficient number of correspondences [8, 35, 41, 17]. Least-squares techniques are used to smooth out noise. In these works, the authors assume that matches are given and are correct. In real applications, however, among the feature correspondences established at the first step, several may be incorrect. These false matches (called *outliers* in terms of robust statistics), sometimes even only one, will completely perturb the motion and structure estimation so that the result will be useless. The reader is referred to [42, 38] and Appendix B for a technique which uses the least-median-squares method to detect false matches. We also mention recent work on recovering motion and structure from long image sequences [26, 4, 5, 44, 37, 30, 2, 36, 34, 20].

The classical approach to motion and structure estimation problem from two given sets of matched image points consists of two stages: (i) using the 8-point algorithm to estimate the 9 essential parameters defined up to a scale factor, which is a linear estimation problem; (ii) refining the motion estimation based on some statistically optimal criteria, which is a nonlinear estimation problem on a five-dimensional space. Unfortunately, the results obtained using this approach are often not satisfactory, especially when the motion is small or when the observed points are close to a degenerate surface (e.g. plane). The problem is that the second stage is very sensitive to the initial guess and that it is very difficult to obtain a precise initial estimate from the first stage. This is because we perform a projection of a set of quantities which are estimated in a space of 8 dimensions, much higher than that of the real space which is five-dimensional [3]. We propose in this paper a novel approach by introducing an intermediate stage which consists in estimating a 3×3 matrix defined up to a scale factor by imposing the *zero-determinant constraint* (the matrix has seven independent parameters, and is known as the fundamental matrix). The idea is to *gradually* project parameters estimated in a high dimensional space onto a *slightly lower* space, namely from 8 dimensions to 7 and finally to 5. The proposed approach has been tested with synthetic and real data, and considerable improvement has been observed for the delicate situations mentioned above. Our conjecture from this work is that the imposition of the constraints arising from projective geometry should be used as an intermediate step in order to obtain reliable 3D Euclidean motion and structure estimation from multiple calibrated images.

For readers who are not interested in the implementation details, they can go directly to Sect. 4 to examine how our new multistage algorithm produces much more reliable results. In the following sections, we present our formulation of the motion and structure from motion problem and describe our technique for determining 3D motion and structure. Besides the introduction of the above-mentioned new stage, our technique differs from the classical

techniques presented in the literature in the work space used. We directly use *pixel* image coordinates, instead of *normalized* image coordinates. We can reasonably assume that the noise levels in both point coordinates are the same if *pixel* coordinates are used, but they are not the same anymore after having been transformed into *normalized* image coordinates because the scales in the two axes are usually not equal (the ratio is approximately 0.7 in our CCD cameras). A criterion based on pixel image coordinates is thus physically more meaningful. (If the ratio is equal to 1, one can, of course, use either pixel or normalized image coordinates.)

2 Notation and Problem Statement

In this section, we formulate the problem we want to solve and describe the epipolar equation which is fundamental in solving motion and structure from motion problems.

2.1 Notation

A camera is described by the widely used pinhole model. The coordinates of a 3-D point $\mathtt{M} = [x, y, z]^T$ in a world coordinate system and its retinal image coordinates $\mathbf{m} = [u, v]^T$ are related by

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbb{P} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} ,$$

where s is an arbitrary scale, and $\mathbb{P}$ is a 3×4 matrix, called the perspective projection matrix. If $\mathbf{x} = [x, y, \cdots]^T$, its augmented vector (adding 1 as its last element) is denoted by $\widetilde{\mathbf{x}}$, i.e. $\widetilde{\mathbf{x}} = [x, y, \cdots, 1]^T$. We thus have $s\widetilde{\mathbf{m}} = \mathbb{P}\widetilde{\mathtt{M}}$.

The matrix $\mathbb{P}$ can be decomposed as

$$\mathbb{P} = \mathbf{A} [\mathbf{R}\ \mathbf{t}] ,$$

where $\mathbf{A}$ is a 3×3 matrix, mapping the normalized image coordinates to the retinal/pixel image coordinates, and $(\mathbf{R}, \mathbf{t})$ is the 3D displacement (rotation and translation) from the world coordinate system to the camera coordinate system. The most general matrix $\mathbf{A}$ can be written as

$$\mathbf{A} = \begin{bmatrix} \alpha_u & c & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} , \tag{1}$$

where the five parameters in $\mathbf{A}$ are known through calibration [10, 39].

The first and second images are respectively denoted by I_1 and I_2. A point $\mathbf{m}$ in the image plane I_i is noted as $\mathbf{m}_i$. The second subscript, if any, will indicate the index of the point in consideration.

2.2 Problem Statement

We consider two perspective images of a single scene, and we want to determine the relation between the two images and the structure of the scene. This can arise from several situations:

- The two images are taken by a moving camera at two different time instants in a static environment. Then the displacement of the camera and the structure of the scene will be estimated.
- The two images are taken by a fixed camera at two different time instants in a dynamic scene. We assume the two images are projections of a single moving rigid object, otherwise a pre-segmentation of images into different regions is necessary. The displacement and structure of the object will be estimated.
- The two images are taken by two cameras either at the same time or at two different instants. In the latter case, we assume the scene is static. The relative location and orientation of the two cameras and the structure of the scene will be estimated.

In either of the above situations, we assume the cameras are calibrated, i.e. their intrinsic parameters, or the $\mathbf{A}$ matrices, are known. Furthermore, since all these problems are mathematically equivalent, we only consider the third situation.

2.3 Epipolar Equation

Considering now the case of two cameras as shown in Fig. 1, where C_1 and C_2 are the optical centers of the cameras. Let the displacement from the first camera to the second be $(\mathbf{R}, \mathbf{t})$. Let $\mathbf{m}_1$ and $\mathbf{m}_2$ be the images of a 3-D point $\mathtt{M}$ on the cameras. Without loss of generality, we assume that $\mathtt{M}$ is expressed in the coordinate frame of the first camera. Under the pinhole model, we have the following two equations:

$$s_1 \widetilde{\mathbf{m}}_1 = \mathbf{A}_1 [\mathbf{I}\ \mathbf{0}] \widetilde{\mathtt{M}} \,, \tag{2}$$

$$s_2 \widetilde{\mathbf{m}}_2 = \mathbf{A}_2 [\mathbf{R}\ \mathbf{t}] \widetilde{\mathtt{M}} \,, \tag{3}$$

where $\mathbf{A}_1$ and $\mathbf{A}_2$ are the intrinsic matrices of the first and second cameras, respectively. Eliminating $\mathtt{M}$, s_1 and s_2 from the above equations, we obtain the following fundamental equation

$$\widetilde{\mathbf{m}}_2^T \mathbf{A}_2^{-T} [\mathbf{t}]_\times \mathbf{R} \mathbf{A}_1^{-1} \widetilde{\mathbf{m}}_1 = 0 \,, \tag{4}$$

where $[\mathbf{t}]_\times$ is an antisymmetric matrix defined by $\mathbf{t}$ such that $[\mathbf{t}]_\times \mathbf{x} = \mathbf{t} \times \mathbf{x}$ for all 3-D vector $\mathbf{x}$ ($\times$ denotes the cross product).

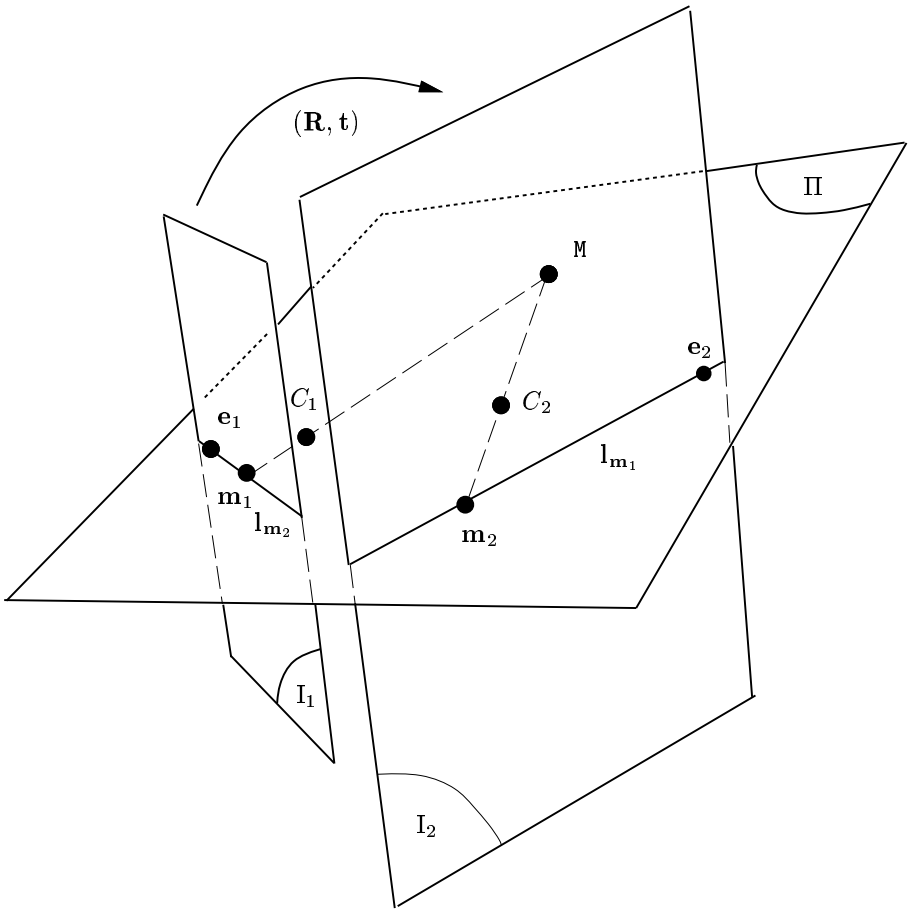

Figure 1: Geometry of motion and structure from motion

Equation (4) is a fundamental constraint underlying any two images if they are perspective projections of one and the same scene. There are two geometric interpretations:

- Equation (4) expresses the fact that four points (C_1, $\mathbf{m}_1$, C_2 and $\mathbf{m}_2$) are coplanar. Indeed, if we consider the coordinate system attached to the second camera, the coordinates of the four points are: $\mathbf{c}_{C_2} = \mathbf{0}$ (by setting $s_2 = 0$ in (3)), $\mathbf{c}_{\mathbf{m}_2} = \mathbf{A}_2^{-1}\mathbf{m}_2$ (by setting $s_2 = 1$ in (3)), $\mathbf{c}_{C_1} = \mathbf{t}$ (by setting $s_1 = 0$ in (2)), $\mathbf{c}_{\mathbf{m}_1} = \mathbf{A}_1^{-1}\mathbf{m}_1$ (by setting $s_1 = 1$ in (2)). The coplanarity implies:

$$(\mathbf{c}_{\mathbf{m}_2} - \mathbf{c}_{C_2}) \cdot [(\mathbf{c}_{C_1} - \mathbf{c}_{C_2}) \times (\mathbf{c}_{\mathbf{m}_1} - \mathbf{c}_{C_2})] = 0 \ ,$$

which gives equation (4).

- Equation (4) can also be interpreted as the point $\mathbf{m}_2$ lying on the epipolar line of $\mathbf{m}_1$. Let

$$\mathbf{F} = \mathbf{A}_2^{-T}[\mathbf{t}]_\times \mathbf{R}\mathbf{A}_1^{-1} ,$$

which is known as the fundamental matrix [7, 24]. The epipolar line of $\mathbf{m}_1$, denoted by $\mathbf{l}_{\mathbf{m}_1}$ in Fig. 1, is the projection of the semi-line $\mathbf{m}_1 C_1$ on the second image, and we have $\mathbf{l}_{\mathbf{m}_1} = \mathbf{F}\tilde{\mathbf{m}}_1$ (i.e. for all point $\mathbf{m}$ on line $\mathbf{l}_{\mathbf{m}_1}$, we have $\tilde{\mathbf{m}}^T\mathbf{F}\tilde{\mathbf{m}}_1 = 0$). The fact that $\mathbf{m}_1$ and $\mathbf{m}_2$ correspond to a single point in space implies that $\mathbf{m}_2$ is on $\mathbf{l}_{\mathbf{m}_1}$, which gives equation (4).

For convenience, we use $\mathbf{p}$ to denote a point in the *normalized* image coordinate system, i.e. $\tilde{\mathbf{p}}_1 = \mathbf{A}_1^{-1}\tilde{\mathbf{m}}_1$, and $\tilde{\mathbf{p}}_2 = \mathbf{A}_2^{-1}\tilde{\mathbf{m}}_2$. Let $\mathbf{E} = [\mathbf{t}]_\times \mathbf{R}$, which is known as the *Essential matrix*. It was introduced by Longuet-Higgins [21], and its property has been studied in the literature [15, 9, 27]. Now, we can write equation (4) as

$$\tilde{\mathbf{p}}_2^T \mathbf{E} \tilde{\mathbf{p}}_1 = 0 . \tag{5}$$

Because the magnitude of $\mathbf{t}$ can never be recovered from two perspective images, we set $\|\mathbf{t}\| = 1$. The relationship between $\mathbf{E}$ and $\mathbf{F}$ is readily described by

$$\mathbf{F} = \mathbf{A}_2^{-T}\mathbf{E}\mathbf{A}_1^{-1} , \quad \text{and} \quad \mathbf{E} = \mathbf{A}_2^T\mathbf{F}\mathbf{A}_1 . \tag{6}$$

Since $[\mathbf{t}]_\times$ is a skew-symmetric matrix, the determinant of matrix $\mathbf{E} = [\mathbf{t}]_\times\mathbf{R}$ must be zero, i.e. matrix $\mathbf{E}$ is rank two. Furthermore, the elements of $\mathbf{E}$ satisfy the following polynomial equation of degree 4 [3, 15]:

$$\|\varepsilon_1 \times \varepsilon_2\|^2 + \|\varepsilon_2 \times \varepsilon_3\|^2 + \|\varepsilon_3 \times \varepsilon_1\|^2 = \frac{1}{4}(\|\varepsilon_1\|^2 + \|\varepsilon_2\|^2 + \|\varepsilon_3\|^2)^2 , \tag{7}$$

where ε_i is the i^{th} row vector of matrix $\mathbf{E}$.

3 The New Multistage Motion Algorithm

In this section, we first recall the 8-point algorithm, which ignores the constraints on the essential parameters, and the nonlinear algorithm, which performs an optimization directly over the five-dimensional motion space. Finally, we show how we can impose the zero-determinant constraint as an intermediate step in order to provide a better initial estimate for the previously mentioned nonlinear algorithm.

3.1 The Linear Criterion

Equation (5) can be rewritten as a linear and homogeneous equation in the 9 unknown coefficients of matrix $\mathbf{E}$:

$$\mathbf{u}^T \boldsymbol{\epsilon} = 0 \,, \tag{8}$$

where

$$\begin{aligned} \mathbf{u} &= [x_1x_2, y_1x_2, x_2, x_1y_2, y_1y_2, y_2, x_1, y_1, 1]^T \\ \boldsymbol{\epsilon} &= [E_{11}, E_{12}, E_{13}, E_{21}, E_{22}, E_{23}, E_{31}, E_{32}, E_{33}]^T \,. \end{aligned}$$

Here $\mathbf{p}_1 = [x_1, y_1]^T$, $\mathbf{p}_2 = [x_2, y_2]^T$, and E_{ij} is the element of $\mathbf{E}$ at row i and column j. If we are given n point matches, by stacking (5), we have the following linear system to solve:

$$\mathbf{U}\boldsymbol{\epsilon} = \mathbf{0} \,,$$

where

$$\mathbf{U} = \begin{bmatrix} \mathbf{u}_1^T \\ \vdots \\ \mathbf{u}_n^T \end{bmatrix} \,.$$

This set of linear homogeneous equations, together with the constraints described at the end of Sect. 2.3, allow us to solve for the motion $(\mathbf{R}, \mathbf{t})$.

The minimum number of point matches is five ($n = 5$) because the rotation has three degrees of freedom and the translation is only determined up to a scale factor. Faugeras and Maybank [9] show that at most ten real solutions are possible in this case, but the algorithm is quite complex. When $n > 5$, we usually have a unique solution, but in some special cases we may have at most three solutions [13, 22, 27]. The algorithm for $n = 6$ is complex, and is not addressed here. For $n = 7$, $\text{rank}(\mathbf{U}) = 7$. Through singular value decomposition, we obtain vectors $\boldsymbol{\epsilon}_1$ and $\boldsymbol{\epsilon}_2$ which span the null space of $\mathbf{U}$. The null space is a linear combination of $\boldsymbol{\epsilon}_1$ and $\boldsymbol{\epsilon}_2$, which correspond to matrices $\mathbf{E}_1$ and $\mathbf{E}_2$, respectively. Because of its homogeneity, the essential matrix is a one-parameter family of matrices $\alpha\mathbf{E}_1 + (1-\alpha)\mathbf{E}_2$. Since the determinant of $\mathbf{E}$ must be null, i.e.

$$\det[\alpha\mathbf{E}_1 + (1-\alpha)\mathbf{E}_2] = 0 \,,$$

we obtain a cubic polynomial in α. The maximum number of real solutions is 3. For each real solution we substitute it into (7) to check if it is satisfied. When data are noisy, none of the solutions satisfies these constraints. If one solution gives a *much smaller* absolute value of the polynomials than the other solutions, it can be considered as the motion between the two images; otherwise, the three solutions are equally feasible, and must all be considered.

If we are given 8 or more matches and ignore the constraints on the essential parameters, we will be able, in general, to determine a unique solution for $\mathbf{E}$, defined up to a scale factor. This can be done by solving the following least-squares problem:

$$\min_{\boldsymbol{\epsilon}} \|\mathbf{U}\boldsymbol{\epsilon}\|^2 \ .$$

Several methods are possible to solve this problem. The first uses a closed-form solution via the linear equations by setting one of the coefficients of $\mathbf{E}$ to 1. The second solves the classical problem:

$$\min_{\boldsymbol{\epsilon}} \|\mathbf{U}\boldsymbol{\epsilon}\|^2 \qquad \text{subject to } \|\boldsymbol{\epsilon}\| = \sqrt{2} \ . \tag{9}$$

The constraint on the norm of $\boldsymbol{\epsilon}$ is derived from the fact that $\mathbf{R}$ is an orthonormal matrix and $\|\mathbf{t}\| = 1$ [6]. The solution is the eigenvector of $\mathbf{U}^T\mathbf{U}$ associated with the smallest eigenvalue. This approach, known as the eight-point algorithm, was proposed by Longuet-Higgins [21] and has been extensively studied in the literature [23, 40, 41, 19]. It has been proven to be very sensitive to noise.

Once we have estimated the essential matrix $\mathbf{E}$, we can recover the motion $(\mathbf{R}, \mathbf{t})$. As $\mathbf{E}^T\mathbf{t} = 0$, the relative location $\mathbf{t}$ is the solution of the following problem:

$$\min_{\mathbf{t}} \|\mathbf{E}^T\mathbf{t}\|^2 \qquad \text{subject to } \|\mathbf{t}\| = 1 \ . \tag{10}$$

Consequently, $\mathbf{t}$ is the unit eigenvector of $\mathbf{E}\mathbf{E}^T$ corresponding to the smallest eigenvalue. If the sign of $\mathbf{E}$ is correct, the ambiguity of the sign of $\mathbf{t}$ can be resolved as follows. Indeed, for j^{th} correspondence $(\mathbf{p}_{1j}, \mathbf{p}_{2j})$, if

$$(\mathbf{t} \times \widetilde{\mathbf{p}}_{2j}) \cdot (\mathbf{E}\widetilde{\mathbf{p}}_{1j}) > 0 \ , \tag{11}$$

then the sign of $\mathbf{t}$ is compatible with the sign of E; otherwise, reverse the sign of $\mathbf{t}$. This can be seen as follows. Let z_{1j} and z_{2j} be, respectively, the depths of their corresponding point in space in the first and second camera coordinate systems. From the pinhole model, we then have $\mathbf{R}(z_{1j}\widetilde{\mathbf{p}}_{1j}) + \mathbf{t} = z_{2j}\widetilde{\mathbf{p}}_{2j}$. This gives

$$z_{2j}(\mathbf{t} \times \widetilde{\mathbf{p}}_{2j}) = z_{1j}(\mathbf{E}\widetilde{\mathbf{p}}_{1j}) \ .$$

As z_{1j} and z_{2j} should be both positive or negative, the two vectors $(\mathbf{t} \times \widetilde{\mathbf{p}}_{2j})$ and $(\mathbf{E}\widetilde{\mathbf{p}}_{1j})$ must have the same direction, and the condition (11) follows. The depth z will be negative if the sign of $\mathbf{E}$ is wrong. The ambiguity of the sign of $\mathbf{E}$ can only be resolved in the reconstruction stage.

We now turn to the problem of estimating the rotation matrix $\mathbf{R}$. As $\mathbf{E} = [\mathbf{t}]_\times\mathbf{R}$ by definition, we find $\mathbf{R}$ by solving

$$\min_{\mathbf{R}} \|\mathbf{E} - [\mathbf{t}]_\times\mathbf{R}\|^2 \qquad \text{subject to } \mathbf{R}^T\mathbf{R} = \mathbf{I} \text{ and } \det(\mathbf{R}) = 1 \ .$$

Since $\mathbf{E} - [\mathbf{t}]_\times \mathbf{R} = (\mathbf{E}\mathbf{R}^T - [\mathbf{t}]_\times)\mathbf{R}$, $\|\mathbf{E} - [\mathbf{t}]_\times \mathbf{R}\|^2 = \|\mathbf{E}\mathbf{R}^T - [\mathbf{t}]_\times\|^2$. The above problem becomes

$$\min_{\mathbf{R}} \sum_{i=1}^{3} \|\mathbf{R}\boldsymbol{\varepsilon}_i - \boldsymbol{\tau}_i\|^2 \qquad \text{subject to } \mathbf{R}^T\mathbf{R} = \mathbf{I} \text{ and } \det(\mathbf{R}) = 1 \,, \tag{12}$$

where $\boldsymbol{\varepsilon}_i$ and $\boldsymbol{\tau}_i$ are the i^{th} *row* vectors of matrices $\mathbf{E}$ and $[\mathbf{t}]_\times$. This can be easily solved using the quaternion representation of 3-D rotations [45].

The ambiguity in the sign of $\mathbf{E}$ can now be resolved. We can use the procedure described in Sect. 3.3 to reconstruct one 3-D point from image pairs. If the z coordinate of the reconstructed point is negative, we must reverse the sign of $\mathbf{t}$, because for points visible to the cameras they must be in front of the cameras (i.e. positive z coordinate). Note that the rotation $\mathbf{R}$ does not need to be recomputed, because inverting both the signs of $\mathbf{t}$ and $\mathbf{E}$ gives the same estimation of $\mathbf{R}$. Although four pairs of $(\mathbf{R}, \mathbf{t})$ can be derived from $\mathbf{E}$, only the pair computed above is physically realizable.

The advantage of the linear criterion is that it leads to an analytic solution. However, we have found that it is quite sensitive to noise, even with a large set of data points. There are two reasons for this:

- We have omitted the constraints on the essential matrix. The elements of $\mathbf{E}$ are not independent from each other.
- The quantity we try to minimize (9) does not have much physical meaning.

3.2 Minimizing the Distances to Epipolar Lines

As was described in Sect. 2.3, $\mathbf{F}\tilde{\mathbf{m}}_1$ represents actually the epipolar line of $\mathbf{m}_1$ in the second image. If $\mathbf{m}_2$ corresponds exactly to $\mathbf{m}_1$, we would expect the distance from $\mathbf{m}_2$ to the epipolar line $\mathbf{F}\tilde{\mathbf{m}}_1$ to be zero. Thus, a natural idea is to use a nonlinear criterion by minimizing:

$$\sum_i d^2(\tilde{\mathbf{m}}_{2i}, \mathbf{F}\tilde{\mathbf{m}}_{1i}) \,,$$

where $d(\tilde{\mathbf{m}}_2, \mathbf{F}\tilde{\mathbf{m}}_1)$ is the Euclidean distance of point $\mathbf{m}_2$ to its epipolar line $\mathbf{F}\tilde{\mathbf{m}}_1$ in the second image. It is given by

$$d(\tilde{\mathbf{m}}_2, \mathbf{F}\tilde{\mathbf{m}}_1) = \frac{\tilde{\mathbf{m}}_2^T \mathbf{F}\tilde{\mathbf{m}}_1}{\sqrt{(\mathbf{F}\tilde{\mathbf{m}}_1)_1^2 + (\mathbf{F}\tilde{\mathbf{m}}_1)_2^2}} \,,$$

where $(\mathbf{F}\tilde{\mathbf{m}}_1)_i$ is the i^{th} component of vector $\mathbf{F}\tilde{\mathbf{m}}_1$, and the distance is *signed*. However, unlike the case of the linear criterion, the two images do not play a symmetric role. To obtain a consistent epipolar geometry, we also consider distances in the first image. This

yields the following criterion:

$$\sum_i (d^2(\tilde{\mathbf{m}}_{2i}, \mathbf{F}\tilde{\mathbf{m}}_{1i}) + d^2(\tilde{\mathbf{m}}_{1i}, \mathbf{F}^T\tilde{\mathbf{m}}_{2i})) \ ,$$

which operates simultaneously in the two images. Using the fact that $\tilde{\mathbf{m}}_2^T\mathbf{F}\tilde{\mathbf{m}}_1 = \tilde{\mathbf{m}}_1^T\mathbf{F}^T\tilde{\mathbf{m}}_2$, it can be rewritten as:

$$\sum_i \left(\frac{1}{(\mathbf{F}\tilde{\mathbf{m}}_{1i})_1^2 + (\mathbf{F}\tilde{\mathbf{m}}_{1i})_2^2} + \frac{1}{(\mathbf{F}^T\tilde{\mathbf{m}}_{2i})_1^2 + (\mathbf{F}^T\tilde{\mathbf{m}}_{2i})_2^2} \right) (\tilde{\mathbf{m}}_{2i}^T\mathbf{F}\tilde{\mathbf{m}}_{1i})^2 \ . \tag{13}$$

Unlike the case of the linear criterion which uses the elements of the essential matrix, we minimize the above functional over the motion parameters. Recall that we deal with calibrated cameras, i.e. $\mathbf{F}$ depends only on $\mathbf{R}$ and $\mathbf{t}$. The rotation is represented by a 3D vector, whose direction is parallel to the rotation axis and whose magnitude is equal to the rotation angle. The translation is represented by its spherical coordinates. Thus, the minimization is carried out over these five unknowns. As the minimization is nonlinear, we use the result of the analytical method as its initial guess.

In the above formulation, we use the *pixel* image coordinates $\mathbf{m}_{ij}$. We can also use the *normalized* image coordinates $\mathbf{p}_{ij}$ with a similar formulation (i.e. replace $\mathbf{F}$ and $\mathbf{m}$ in (13) by $\mathbf{E}$ and $\mathbf{p}$). We have implemented both criteria. Experiments have shown that better results were obtained using *pixel* image coordinates, i.e. (13), than using *normalized* image coordinates. This is because points are usually extracted in pixel images, but not in normalized images. We can reasonably assume that the noise levels in both point coordinates are the same if *pixel* coordinates are used, but they are not the same anymore after having been transformed into *normalized* image coordinates because the scales in the two axes are usually not equal (the ratio is approximately 0.7 in our CCD cameras). Hence, the criterion (13) is physically more meaningful than using normalized image coordinates.

3.3 3D Reconstruction

Once we know the motion $(\mathbf{R}, \mathbf{t})$, given a match $(\mathbf{m}_1, \mathbf{m}_2)$, it is a straightforward matter to formulate a linear least-squares to estimate the 3D coordinates $\mathtt{M}$ by eliminating s_1 and s_2 from (2) and (3). Let $\mathbf{m}_1 = [u_1, v_1]^T$, $\mathbf{m}_2 = [u_2, v_2]^T$, and $\mathbf{B}_2 = \mathbf{A}_2\mathbf{R}$, then we have

$$\begin{bmatrix} \mathbf{a}_1^T - u_1\mathbf{a}_3^T \\ \mathbf{a}_2^T - v_1\mathbf{a}_3^T \\ \mathbf{b}_1^T - u_2\mathbf{b}_3^T \\ \mathbf{b}_2^T - v_2\mathbf{b}_3^T \end{bmatrix} \mathtt{M} = \begin{bmatrix} 0 \\ 0 \\ (u_2\mathbf{c}_3 - \mathbf{c}_1)^T\mathbf{t} \\ (v_2\mathbf{c}_3 - \mathbf{c}_2)^T\mathbf{t} \end{bmatrix} \ , \qquad \text{or } \mathbf{Z}\mathtt{M} = \mathbf{z} \ ,$$

where $\mathbf{a}_i^T$, $\mathbf{b}_i^T$ and $\mathbf{c}_i^T$ are respectively the i^{th} row of matrices $\mathbf{A}_1$, $\mathbf{B}_2$ and $\mathbf{A}_2$. The solution is given by $\hat{\mathtt{M}} = (\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T\mathbf{z}$. A more elaborate technique consists in minimizing the distance

between the back-projection of the 3D reconstruction and the observed image point, that is

$$\hat{\mathtt{M}} = \arg\min_{\mathtt{M}} \left(\|\mathbf{m}_1 - \mathbf{h}_1(\mathbf{a}, \mathtt{M})\|^2 + \|\mathbf{m}_2 - \mathbf{h}_2(\mathbf{a}, \mathtt{M})\|^2 \right) ,$$

where $\mathbf{h}_1(\mathbf{a}, \mathtt{M})$ and $\mathbf{h}_2(\mathbf{a}, \mathtt{M})$ are the camera projection functions corresponding to (2) and (3), respectively.

3.4 Statistically Optimal Motion and Structure Estimation

We are given n point matches $\{(\mathbf{m}_{1j}, \mathbf{m}_{2j}) | j = 1, \ldots, n\}$. Each point is assumed to be corrupted by additive independent Gaussian noise, i.e.

$$\mathbf{m}_{ij} = \bar{\mathbf{m}}_{ij} + \boldsymbol{\eta}_{ij} \quad \text{for } i = 1, 2 \text{ and } j = 1, \ldots, n, \tag{14}$$

where $\bar{\mathbf{m}}_{ij}$ is the *ideal* point position if noise did not exist, and $\boldsymbol{\eta}_{ij}$ is a Gaussian random vector of mean $\mathbf{0}$ and covariance matrix $\boldsymbol{\Lambda}_{ij}$, i.e. $\boldsymbol{\eta}_{ij} \sim N(\mathbf{0}, \boldsymbol{\Lambda}_{ij})$. The noise terms in different points are independent, i.e. $E[\boldsymbol{\eta}_{ij}\boldsymbol{\eta}_{kl}^T] = \mathbf{O}$ for $j \neq l$. Let us denote the observed points by $\mathbf{x}$, i.e.

$$\mathbf{x} = [\mathbf{m}_{11}^T, \mathbf{m}_{21}^T, \ldots, \mathbf{m}_{1j}^T, \mathbf{m}_{2j}^T, \ldots, \mathbf{m}_{1n}^T, \mathbf{m}_{2n}^T]^T ,$$

which is a vector of $4n$ dimensions.

Let $\mathbf{a} = [\mathbf{r}^T, \boldsymbol{\phi}^T]^T$ be the 5-D vector composed of three parameters representing the rotation between the two images and two parameters representing the translation (see Sect. 3.2). Let $\mathtt{M}_j$ be the 3-D vector corresponding to the j^{th} point expressed in the coordinate system associated with the first camera. The motion and structure parameters are then represented by a vector of $(5 + 3n)$ dimensions, denoted by

$$\boldsymbol{\theta} = [\mathbf{a}^T, \mathtt{M}_1^T, \ldots, \mathtt{M}_j^T, \ldots, \mathtt{M}_n^T]^T .$$

The maximum likelihood (ML) estimate, $\hat{\boldsymbol{\theta}}$, of the parameter vector $\boldsymbol{\theta}$ is given by the value of $\boldsymbol{\theta}$ which makes the observed data $\mathbf{x}$ most likely, that is,

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \log[P(\mathbf{x}|\boldsymbol{\theta})] , \tag{15}$$

where $P(\mathbf{x}|\boldsymbol{\theta})$ is the conditional probability density of the observed data $\mathbf{x}$ given the parameter vector $\boldsymbol{\theta}$.

Under the assumption that the data is corrupted by independent Gaussian noise, we have

$$P(\mathbf{x}|\boldsymbol{\theta}) = C \exp\Big[-\frac{1}{2} \sum_{i=1}^{2} \sum_{j=1}^{n} (\mathbf{m}_{ij} - \mathbf{h}_i(\mathbf{a}, \mathtt{M}_j))^T \boldsymbol{\Lambda}_{ij}^{-1} (\mathbf{m}_{ij} - \mathbf{h}_i(\mathbf{a}, \mathtt{M}_j)) \Big] ,$$

where C is a constant term

$$C = (2\pi)^{-n}\Big(\prod_{i=1}^{2}\prod_{j=1}^{n}|\mathbf{\Lambda}_{ij}|^{-1/2}\Big) ,$$

and $\mathbf{h}_1(\mathbf{a}, \mathtt{M}_j)$ and $\mathbf{h}_2(\mathbf{a}, \mathtt{M}_j)$ are the camera projection functions corresponding to (2) and (3), respectively. It is then clear that the solution to the above ML-estimation problem (15) is equivalent to the following weighted nonlinear least-squares formulation:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{2}\sum_{j=1}^{n}(\mathbf{m}_{ij} - \mathbf{h}_i(\mathbf{a}, \mathtt{M}_j))^T \mathbf{\Lambda}_{ij}^{-1}(\mathbf{m}_{ij} - \mathbf{h}_i(\mathbf{a}, \mathtt{M}_j))\Big] , \tag{16}$$

which is actually the sum of squared Mahalanobis distances. Due to the nonlinear nature of perspective projection, the solution to the above problem demands the use of numerical nonlinear minimization technique such as the Levenberg-Marquardt algorithm implemented in the `MINPACK` library [28]. An initial guess on the motion and structure is required, which can be obtained by using the techniques described previously.

The exact value of the covariance matrix $\mathbf{\Lambda}_{ij}$ is very difficult to obtain in practice. It depends on the point detector used, and on the image intensity variation in the neighborhood of the feature point. However, qualitatively, we expect, and it is confirmed by our experience, that the noise is reasonably isotropic in the image plane and identically distributed. Thus, we can assume

$$\mathbf{\Lambda}_{ij} = \sigma^2 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} , \tag{17}$$

where σ is called the *noise level* which depends on the quality of the point detector used. We do not need to know the noise level, because the minimization is not affected by a multiplication of a constant value. From this assumption, the problem (16) can be simplified as

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{2}\sum_{j=1}^{n} \|\mathbf{m}_{ij} - \mathbf{h}_i(\mathbf{a}, \mathtt{M}_j)\|^2 . \tag{18}$$

The solution to either (16) or (18) requires a numerical minimization to be performed in a $(5+3n)$-D space, which is huge for a large n. By examining the above formulation, we find that the structure parameter $\mathtt{M}_j$ is only involved in two terms. We can thus "separate" the motion estimation from the structure estimation [41], namely

$$\hat{\boldsymbol{\theta}} = \arg\min_{\mathbf{a}} \sum_{j=1}^{n} \Big[\min_{\mathtt{M}_j} \big(\|\mathbf{m}_{1j} - \mathbf{h}_1(\mathbf{a}, \mathtt{M}_j)\|^2 + \|\mathbf{m}_{2j} - \mathbf{h}_2(\mathbf{a}, \mathtt{M}_j)\|^2\big)\Big] . \tag{19}$$

That is, we conduct an outer minimization over $\mathbf{a}$ which is 5-dimensional, and n *independent* inner minimizations over $\mathtt{M}_j$ which is 3-dimensional.

3.5 Imposing the Zero-Determinant Constraint to Refine the Initial Motion Estimate

The two problems described in Sect. 3.2 and Sect. 3.4 are both highly nonlinear, and their solutions are very sensitive to the initial guess. The initial guess is obtained by projecting the essential parameters onto the five-dimensional motion space. Unfortunately, the estimation of the essential parameters as described in Sect. 3.1 is very sensitive to data noise, especially when motion is small and the space points are close to a degenerate surface such as a plane. One major reason is that we have ignored the constraints which exist on the essential parameters and have used 8 parameters instead of 5 (i.e. three redundant parameters). Here, we propose to impose the zero-determinant (rank-2) constraint, and to estimate 7 parameters before projecting onto the motion space.

Indeed, there are only 7 independent parameters in a rank-2 matrix defined up to a scale factor (the scale factor and the rank-2 constraint remove two free parameters), and the fundamental matrix in the context of two uncalibrated images [24, 25] has exactly the same properties. There are several possible parameterizations for such a matrix, e.g. one can express one row (or column) of the fundamental matrix as the linear combination of the other two rows (or columns). The following parameterization

$$\mathbf{F} = \begin{bmatrix} a & b & -ax_1 - by_1 \\ c & d & -cx_1 - dy_1 \\ -ax_2 - cy_2 & -bx_2 - dy_2 & (ax_1 + by_1)x_2 + (cx_1 + dy_1)y_2 \end{bmatrix} \tag{20}$$

expresses of course a matrix of rank 2, because both the third row and column are the combinations of the other two rows and columns. Furthermore, there is a nice geometric interpretation. The parameters (x_1, y_1) and (x_2, y_2) are the coordinates of the two epipoles $\mathbf{e}_1$ (projection of the optical center of the second camera in the first camera) and $\mathbf{e}_2$ (projection of the optical center of the first camera in the second camera) (see Fig. 1). The remaining four parameters (a, b, c, d) define the relationship between the orientations of the two pencils of epipolar lines. To take into account the fact that the matrix is defined only up to a scale factor, the matrix is normalized by diving the four elements (a, b, c, d) by the largest in absolute value.

The seven parameters, $(x_1, y_1, x_2, y_2$, and three among $a, b, c, d)$, are estimated by minimizing the sum of distances between points and their epipolar lines. That is, we minimize the same objective function as the one (13) described in Sect. 3.2, only the minimization is conducted over the above *7-dimensional parameter space*, instead of the five-dimensional motion space. The minimization is nonlinear, and we use the matrix estimated in (9) as the initial guess. The determinant of that matrix, denoted by $\mathbf{M}$ for clarity, is in general not

equal to zero. Let

$$\mathbf{M} = \mathbf{U}\mathbf{S}\mathbf{V}^T$$

be the singular value decomposition of matrix $\mathbf{M}$, where $\mathbf{S} = \mathrm{diag}(s_1, s_2, s_3)$ is a diagonal matrix satisfying $s_1 \geq s_2 \geq s_3 \geq 0$ (s_i is the i^{th} singular value), and $\mathbf{U}$ and $\mathbf{V}$ are orthogonal matrices. Then, it can be shown that

$$\mathbf{F} = \mathbf{U}\hat{\mathbf{S}}\mathbf{V}^T \tag{21}$$

with $\hat{\mathbf{S}} = \mathrm{diag}(s_1, s_2, 0)$ is the matrix of rank-2 which minimizes the Frobenius norm of $\mathbf{M} - \mathbf{F}$ [11]. It is easy to verify that

$$\mathbf{F}\tilde{\mathbf{e}}_1 = \mathbf{0} \quad \text{and} \quad \mathbf{F}^T\tilde{\mathbf{e}}_2 = \mathbf{0}\ . \tag{22}$$

Therefore, $\tilde{\mathbf{e}}_1 = [e_{11}, e_{12}, e_{13}]^T$ and $\tilde{\mathbf{e}}_2 = [e_{21}, e_{22}, e_{23}]^T$ are equal to the last column of $\mathbf{V}$ and $\mathbf{U}$, respectively. From them, we have

$$x_i = e_{i1}/e_{i3} \quad \text{and} \quad y_i = e_{i2}/e_{i3} \qquad \text{for } i = 1, 2.$$

In turn, the four remaining elements (a, b, c, d) can be computed from $\mathbf{F}$.

Note that this intermediate stage can be applied to normalized image coordinates as well as pixel image coordinates.

3.6 Summary of the New Multistage Algorithm

We now summarize the main steps of our multistage algorithm:

Step 1: Estimate the essential parameters with 8-point algorithm (9). The obtained matrix is denoted by $\mathbf{E}_1$.

Step 2: Estimate a rank-2 matrix, denoted by $\mathbf{E}_2$, from $\mathbf{E}_1$ using (21), and compute the seven parameters from $\mathbf{E}_2$.

Step 3: Refine the seven parameters by minimizing the sum of squared distances between points and their epipolar lines, i.e. the objective function (13). The obtained matrix is denoted by $\mathbf{E}_3$.

Step 4: Estimate the motion parameters $\mathbf{t}$ and $\mathbf{R}$ from $\mathbf{E}_3$ using (10) and (12), respectively.

Step 5: Refine the motion parameters by minimizing the sum of squared distances between points and their epipolar lines, i.e. the objective function (13).

Step 6: Reconstruct the corresponding 3D points as described in Sect. 3.3.

Step 7: Refine the motion and structure estimate by using the statistically optimal criterion (19).

The nonlinear minimization in steps 3, 5, and 7 is done with the Levenberg-Marquardt algorithm implemented in the `Minpack` library [28].

If we bypass steps 2 and 3, we have a classical 2-stage algorithm.

4 Experimental Results

In this section, we first describe our Monte-Carlo simulations to show that our new multistage algorithm yields much more reliable results than the classical one when the level of noise in data points is high or when data points are located close to a degenerate configuration. We then present a set of real data with which the classical algorithm does not work while our does. Finally, we give another set of Monte-Carlo simulations to show that better results can be obtained if we work directly with pixel coordinates rather than normalized image coordinates, because points are usually extracted from pixel images.

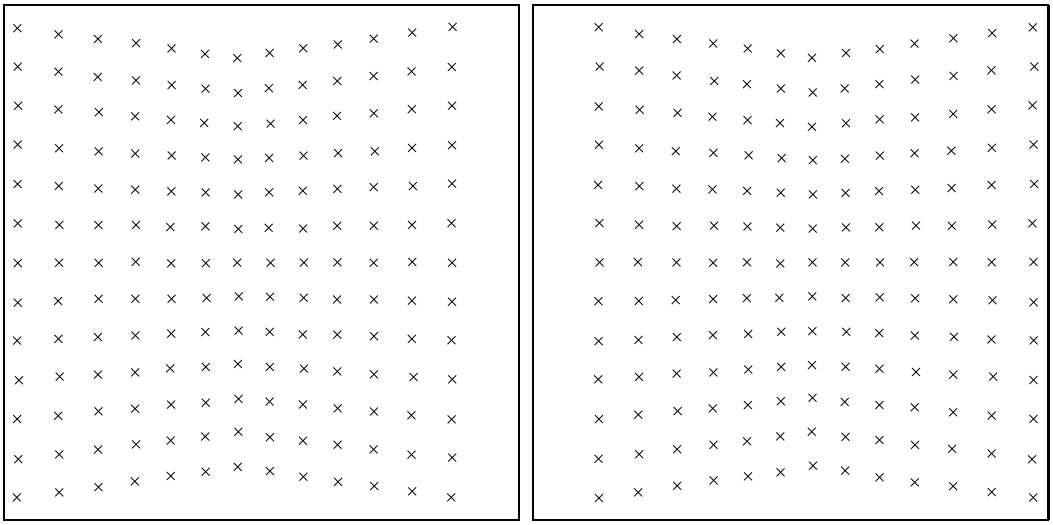

Figure 2: Images of two planar grids hinged together with $\theta = 45°$. Gaussian noise of $\sigma = 0.5$ has been added to each grid point

4.1 Computer Simulated Data

We use the same configuration as that described in [18]. The object is composed of two planar grids which are hinged together with angle $\pi - \theta$. When $\theta = 0$, the object is planar, which is a degenerate configuration for the algorithms considered in this paper. Each grid is of size 180×360 units^2. The object is placed in the scene with a distance of 530 units from the camera. The two images have the same intrinsic parameters: $\alpha_u = \alpha_v = 600$, $u_0 = v_0 = 255$,

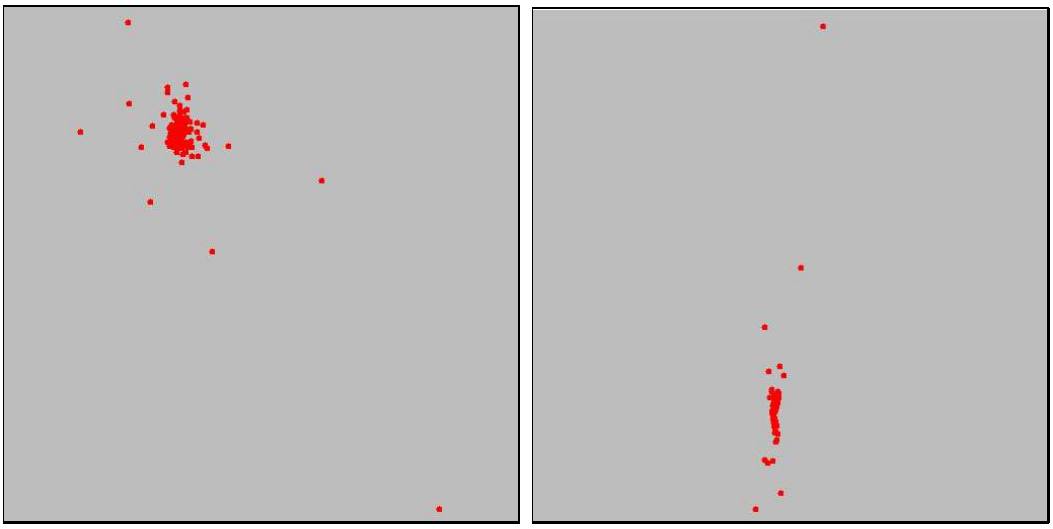

Figure 3: 3D reconstruction of the images shown in Fig. 2 with the 2-stage algorithm: Front and top views

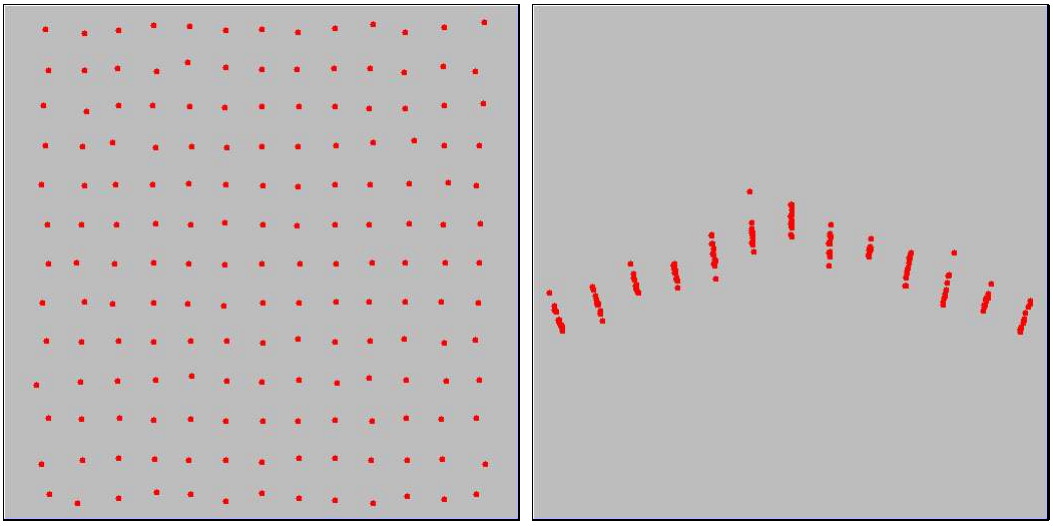

Figure 4: 3D reconstruction of the images shown in Fig. 2 with the 3-stage algorithm: Front and top views

and $c = 0$. They differ by a pure lateral translation: $\mathbf{t} = [-40, 0, 0]^T$, and $\mathbf{R} = \mathbf{I}$. Small lateral motion is difficult for motion estimation because rotation and translation can be confused. The grid points are used as feature points. The x- and y-coordinates of each grid point are perturbed by independent random Gaussian noise of mean 0 and standard deviation σ pixels.

A pair of images with $\theta = 45°$ and $\sigma = 0.5$ pixels is shown in Fig. 2. The motion estimate given by the 2-stage algorithm is: $\mathbf{r} = [-7.981238e-05, -7.961793e-02, 3.779707e-04]^T$ (in radians) for rotation (which should be $[0, 0, 0]^T$), and $\mathbf{t} = [2.091024e-01 - 3.089894e-02 - 9.774055e-01]^T$ for translation (which should be $[-1, 0, 0]^T$). The corresponding 3D reconstruction is shown in Fig. 3. Clearly, the 2-stage algorithm fails. When we apply the 3-stage algorithm to the same data, we obtain $\mathbf{r} = [-6.969567e-04, -1.785441e-03, -2.330740e-04]^T$ for rotation, and $\mathbf{t} = [-9.999643e-01, -8.318301e-03, 1.468760e-03]^T$ for translation. The corresponding 3D reconstruction is shown in Fig. 4. As can be observed, quite reasonable result has been obtained with our new multistage algorithm, taking into account the fact that the object is close to a plane surface.

Now we provide more systematic and statistic results. We vary the angle θ from 10° to 90° with an interval of 10°. We also vary the level of the Gaussian noise added to the grid points. The standard deviation σ varies from 0.25 pixels to 2.0 pixels with an interval of 0.25 pixels. For each θ and each σ, we add 100 times independent noise to the grid points. For each set of noisy data, we apply the 2-stage algorithm and our multistage algorithm. If the estimated translation vector and the true one (i.e. $[-1, 0, 0]^T$) form an angle larger than 45°, then the algorithm is considered to have failed for this set of data. Among 100 trials for each θ and each σ, we count the number of times that the algorithm succeeds. The result is shown in Table 1. For a more direct perception of the difference of the two algorithms in performance, we show in Fig. 5 the curves of the number of successes with respect to various noise levels when θ is fixed at 60° and 90°, respectively. In Fig. 6, we show the curves with respect to various angles θ when the noise level σ is fixed at 0.5 pixels. A general rule is that the number of success decreases when the angle θ approaches to 0° and when the noise level σ increases. In all cases, our new multistage algorithm outperforms the 2-stage algorithm. The 3-stage algorithm gives much more reliable motion and structure estimate when the points are close to a planar surface (a degenerate configuration for the motion algorithms considered here) and when data points are heavily corrupted by noise. A final point is that the two algorithms give the same result when they both converge. This is not surprising because the same statistically optimal technique is used in the last stage.

4.2 Real Data

In Fig. 7, we show a real image pair taken in a rock scene. We have overlayed on the images the point matches automatically established by the techniques described in [42] and Appendix B. When the 2-stage algorithm applies to this set of matches, the motion estimate is $\mathbf{r} = [-2.566072e-02, 1.801078e-03, 2.640767e-02]^T$ for rotation, and $\mathbf{t} = [1.986539e-02, 3.913866e-02, -9.990363e-01]^T$ for translation. We do not have the